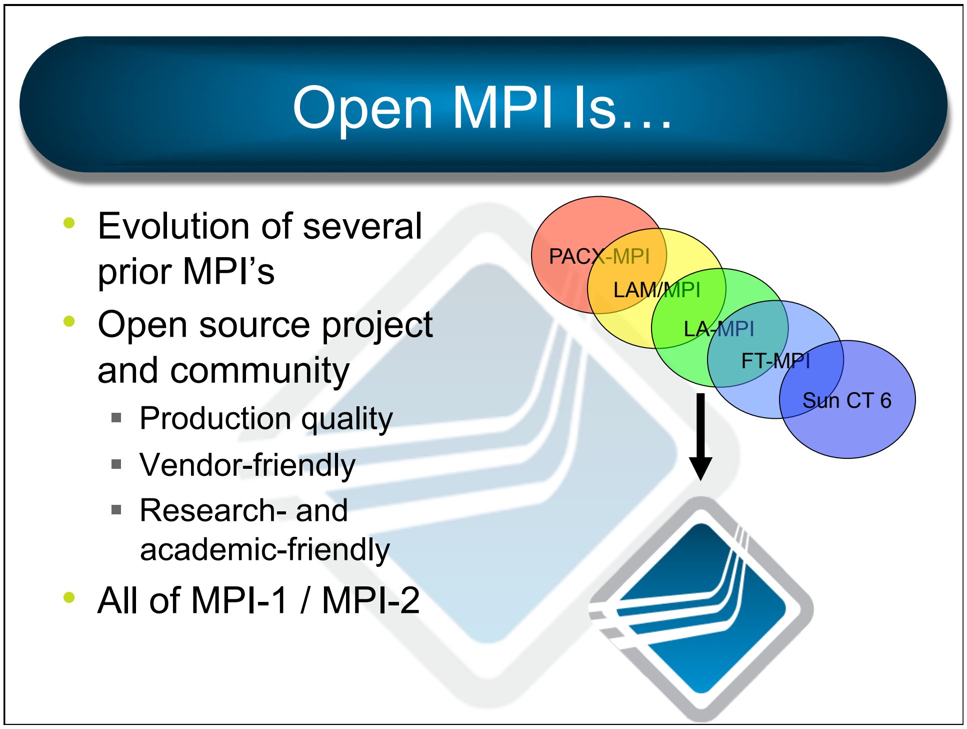
Open MPI State of the Union Community Meeting SC'09

Jeff Squyres George Bosilca

CISCO
ICL
UT

Agenda

- Open MPI Project / Community
- Current Status: v1.3.4 → v1.4
- Next Release Series: v1.5
- Upcoming Challenges
- HPC Community Feedback



Open MPI Is…

- Evolution of several prior MPI's
- Open source project and community
  - Production quality
  - Vendor-friendly
  - Research- and academic-friendly
- All of MPI-1 / MPI-2

PACX-MPI

LAM/MPI

LA-MPI

FT-MPI

Sun CT 6

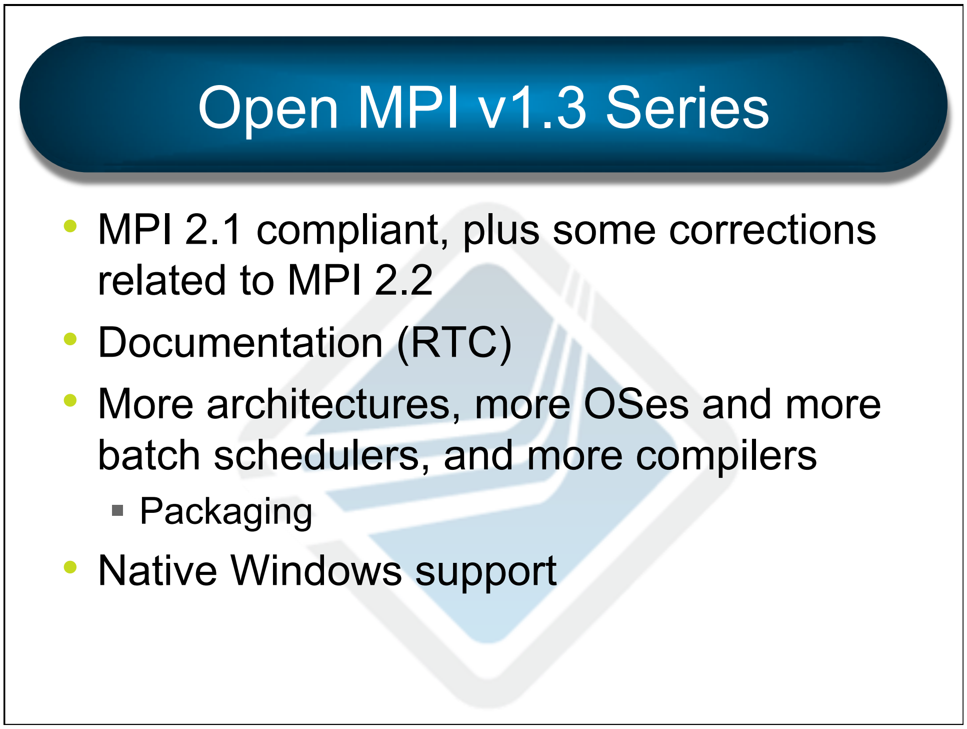
16 Members, 9 Contributors, 2 Partners

pervasivetechnologylabs
AT INDIANA UNIVERSITY
ICL UT
Mellanox TECHNOLOGIES
IBM
Myricom
librato
UBC
CHEMNITZ UNIVERSITY OF TECHNOLOGY
CISCO
Los Alamos NATIONAL LABORATORY
OAK RIDGE National Laboratory
Sandia National Laboratories
H L R S
Sun microsystems
Platform
INRIA
CS@UH
Chelsio Communications
QLOGIC
Voltaire The Grid Backbone
ZIH
AIST
coverity
absoft

Current Status: v1.3.4

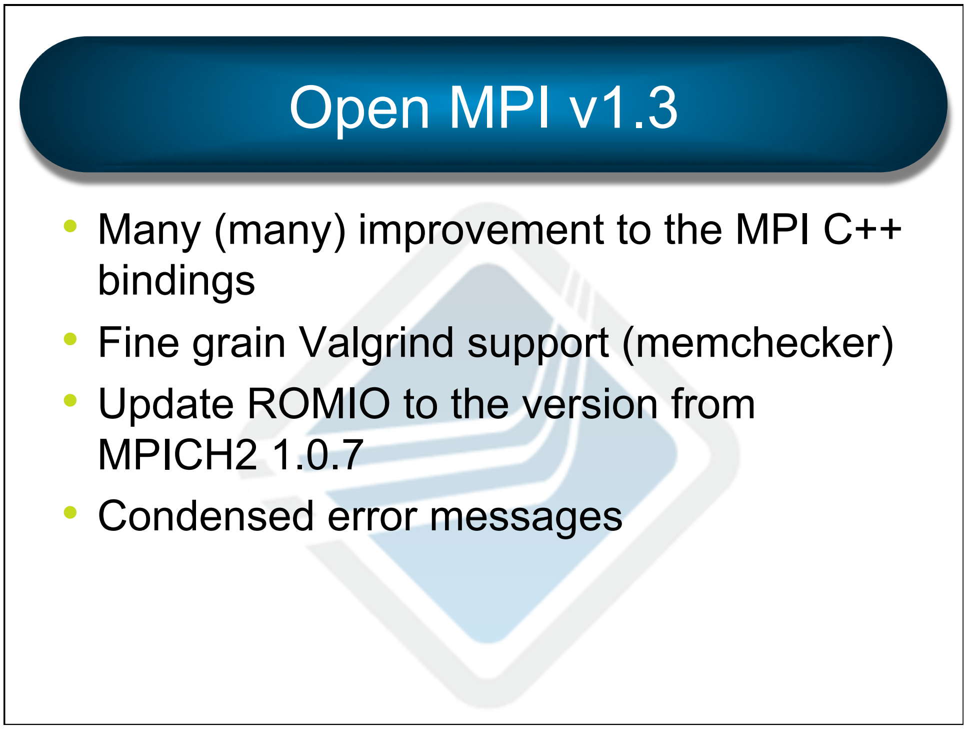
Open MPI v1.3.4

- Release Managers:
  - Brad Benton (IBM)
  - George Bosilca (UTK)
- Gate Keepers
  - Ralph Castain (LANL/Cisco)
  - Jeff Squyres (Cisco)
- Expected as soon as possible after SC’09!



Open MPI v1.3 Series

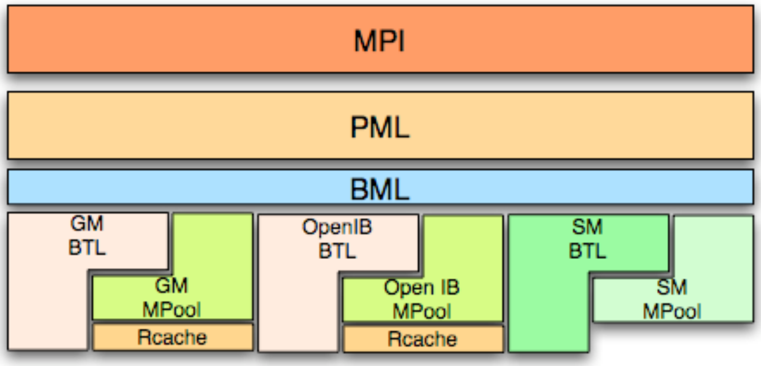
- MPI 2.1 compliant, plus some corrections related to MPI 2.2
- Documentation (RTC)
- More architectures, more OSes and more batch schedulers, and more compilers
  - Packaging
- Native Windows support



Open MPI v1.3

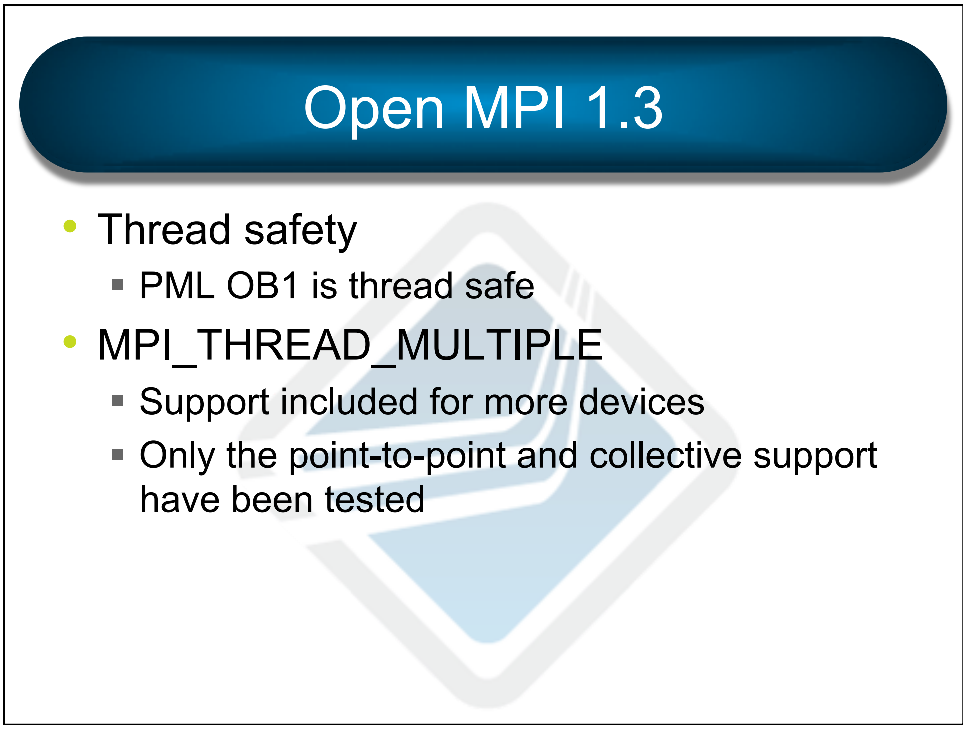
- Many (many) improvement to the MPI C++ bindings
- Fine grain Valgrind support (memchecker)
- Update ROMIO to the version from MPICH2 1.0.7
- Condensed error messages

Open MPI v1.3



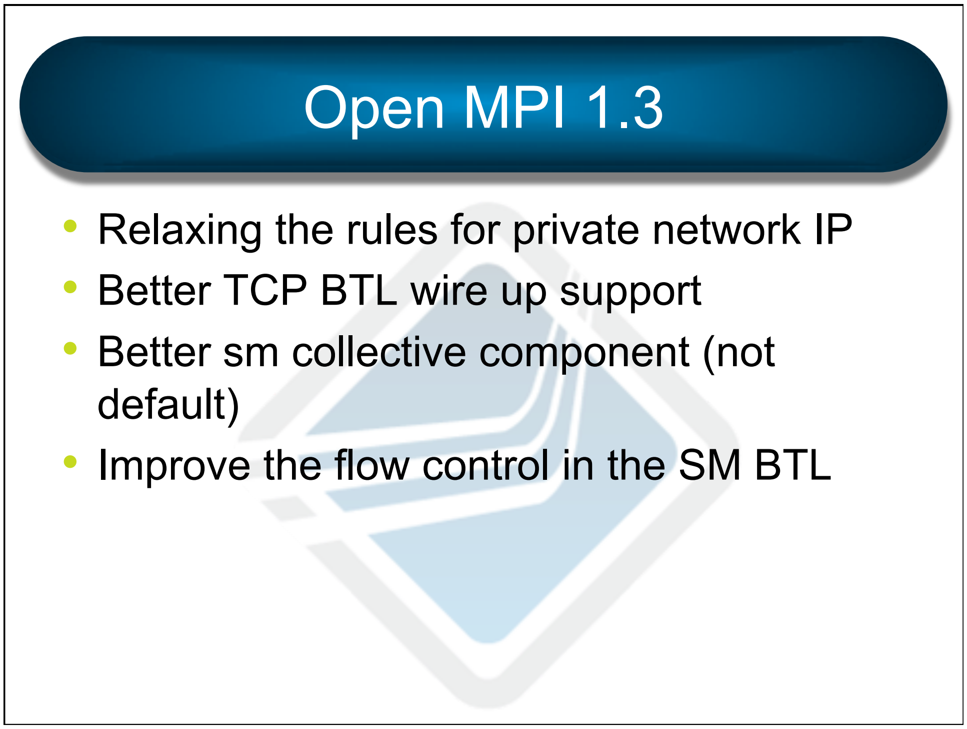
Open MPI 1.3

- Upper level
  - Process affinity options to mpirun: npersocket, npernode, loadbalance, bind-to-socket
  - Progress meter for launching large jobs (orte_report_launch_progress)
- ABI compatibility between versions: as long as the MPI doesn’t change your linked applications will run independent on the Open MPI version available (starting with the 1.3)
- New frameworks
  - The notifier framework



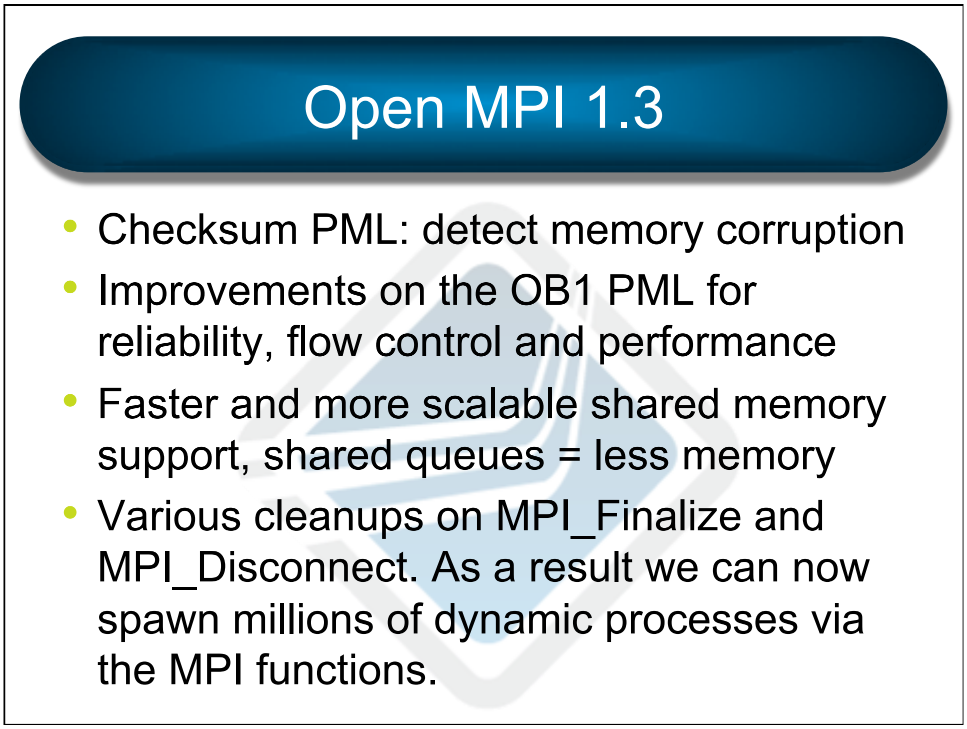
Open MPI 1.3

- Thread safety
  - PML OB1 is thread safe
- MPI_THREAD_MULTIPLE
  - Support included for more devices
  - Only the point-to-point and collective support have been tested



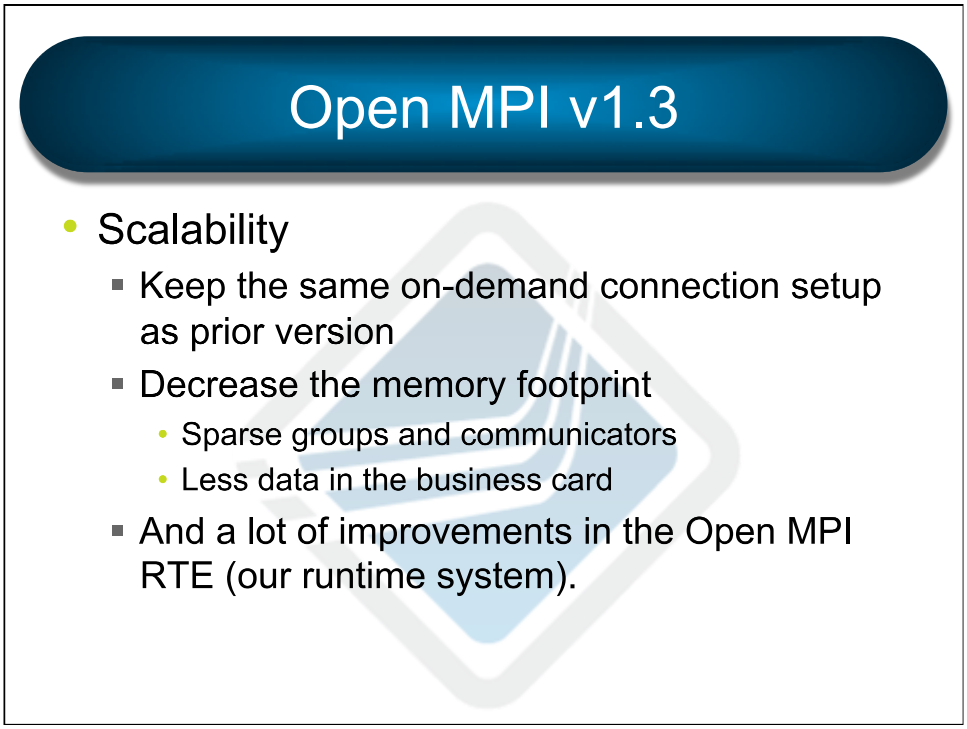
Open MPI 1.3

- Relaxing the rules for private network IP
- Better TCP BTL wire up support
- Better sm collective component (not default)
- Improve the flow control in the SM BTL



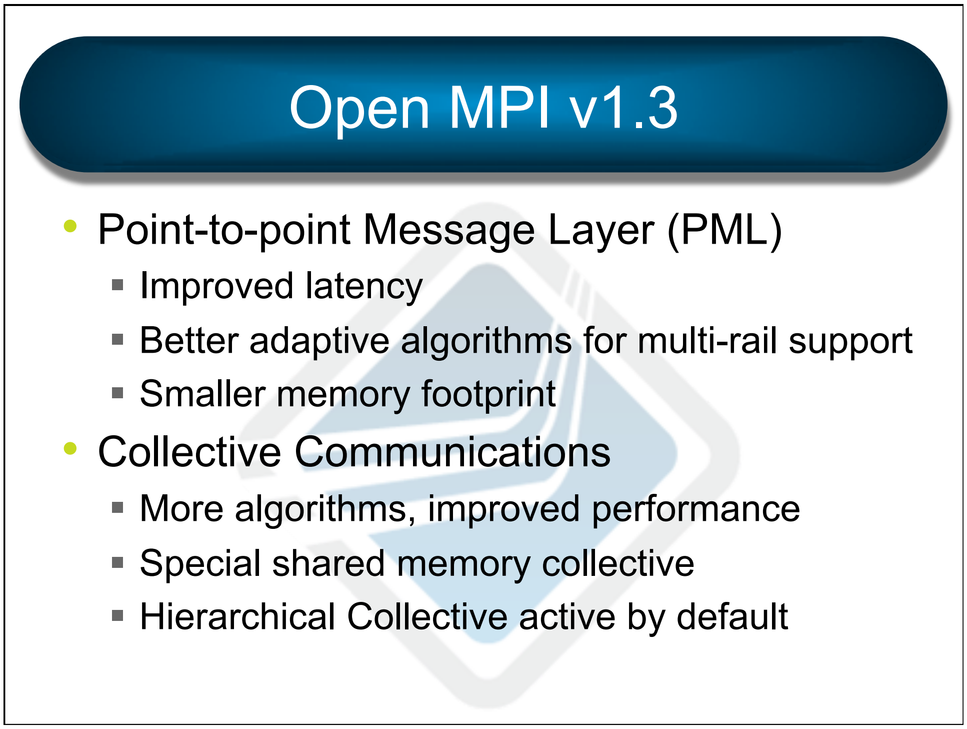
Open MPI 1.3

- Checksum PML: detect memory corruption
- Improvements on the OB1 PML for reliability, flow control and performance
- Faster and more scalable shared memory support, shared queues = less memory
- Various cleanups on MPI_Finalize and MPI_Disconnect. As a result we can now spawn millions of dynamic processes via the MPI functions.



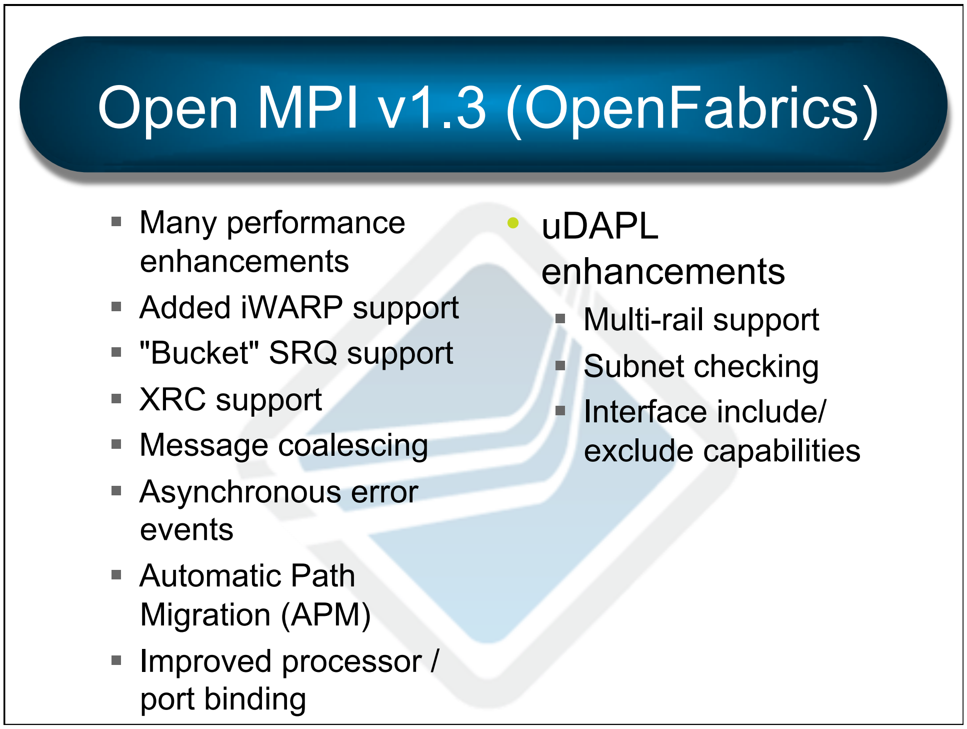
Open MPI v1.3

- Scalability
  - Keep the same on-demand connection setup as prior version
  - Decrease the memory footprint
    - Sparse groups and communicators
    - Less data in the business card
  - And a lot of improvements in the Open MPI RTE (our runtime system).



Open MPI v1.3

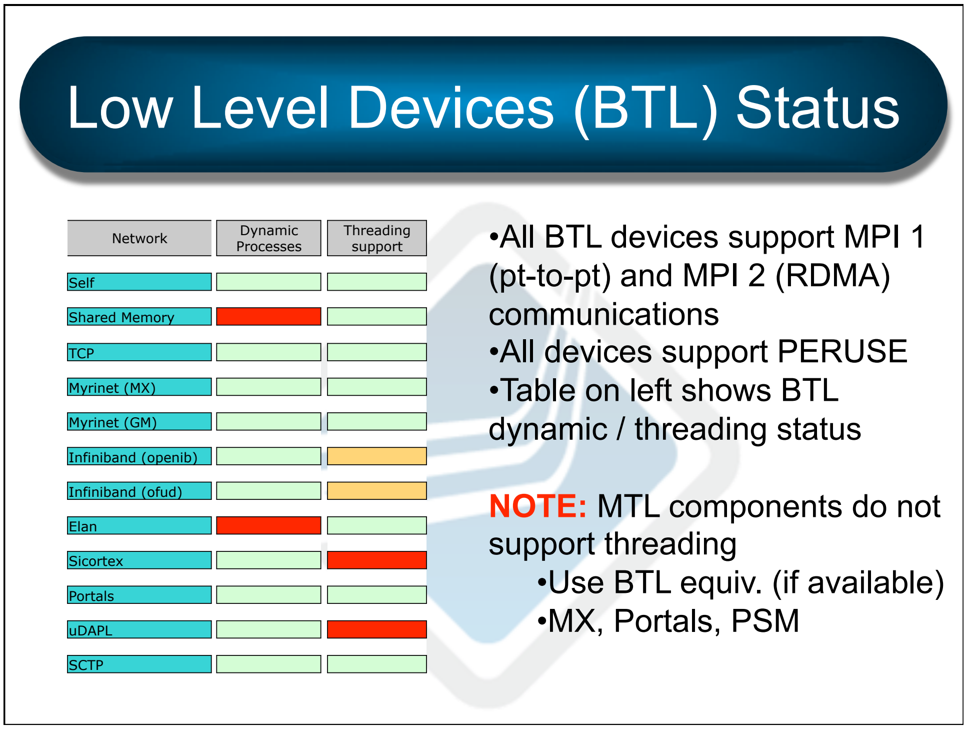
- Point-to-point Message Layer (PML)
  - Improved latency
  - Better adaptive algorithms for multi-rail support
  - Smaller memory footprint
- Collective Communications
  - More algorithms, improved performance
  - Special shared memory collective
  - Hierarchical Collective active by default



Open MPI v1.3 (OpenFabrics)

- Many performance enhancements
- Added iWARP support
- "Bucket" SRQ support
- XRC support
- Message coalescing
- Asynchronous error events
- Automatic Path Migration (APM)
- Improved processor / port binding

- uDAPL enhancements
  - Multi-rail support
  - Subnet checking
  - Interface include/ exclude capabilities



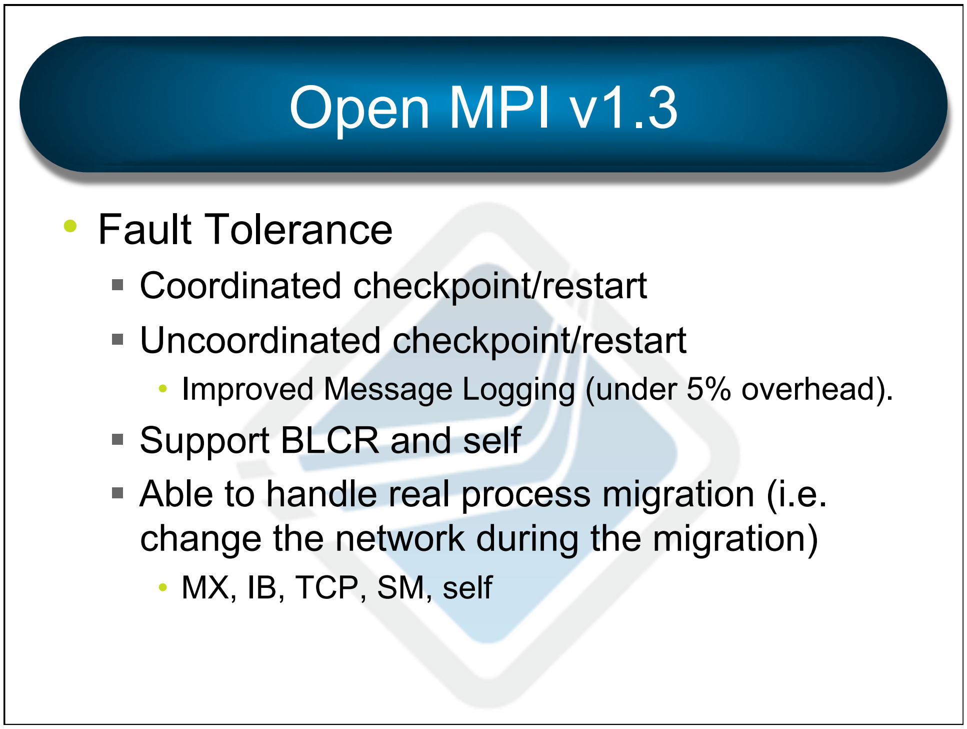
Low Level Devices (BTL) Status

Network	Dynamic Processes	Threading support
Self	🟩	🟩
Shared Memory	🟥	🟩
TCP	🟩	🟩
Myrinet (MX)	🟩	🟩
Myrinet (GM)	🟩	🟩
Infiniband (openib)	🟩	🟧
Infiniband (ofud)	🟩	🟧
Elan	🟥	🟩
Sicortex	🟩	🟥
Portals	🟩	🟩
uDAPL	🟩	🟥
SCTP	🟩	🟩

- All BTL devices support MPI 1 (pt-to-pt) and MPI 2 (RDMA) communications
- All devices support PERUSE
- Table on left shows BTL dynamic / threading status

NOTE: MTL components do not support threading

  - Use BTL equiv. (if available)
  - MX, Portals, PSM



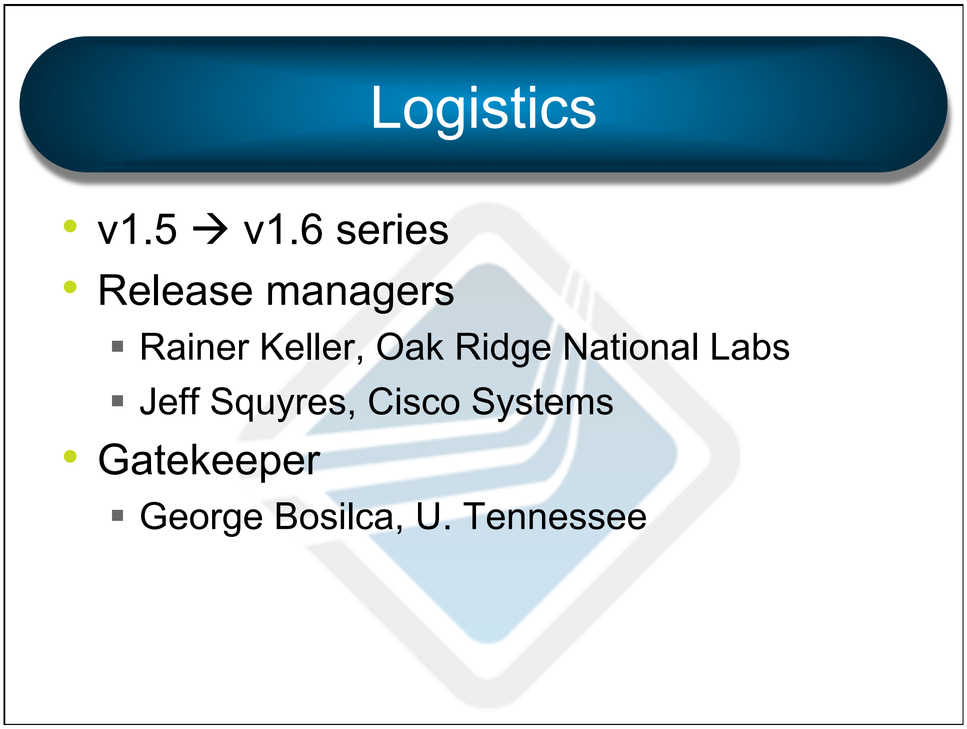
Open MPI v1.3

- Fault Tolerance
  - Coordinated checkpoint/restart
  - Uncoordinated checkpoint/restart
    - Improved Message Logging (under 5% overhead).
  - Support BLCR and self
  - Able to handle real process migration (i.e. change the network during the migration)
    - MX, IB, TCP, SM, self

Version Numbering

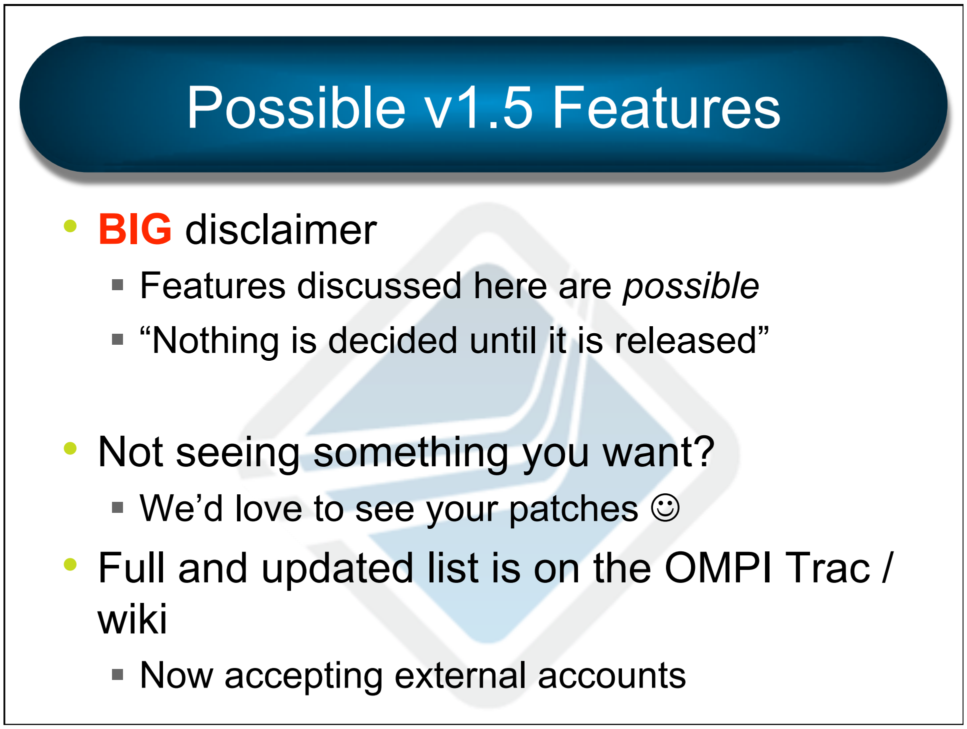
- We have [at least] 2 competing forces in Open MPI:
  - desire to release new features quickly. Fast is good.
  - desire to release based on production quality. Slow is good.
- Open MPI will have two concurrent release series:
  - "Super stable": for production users (even minor)
  - "Feature driven”: not that bleeding edge (odd minor)
  - Trunk for everybody else …

Next Release Series: v1.5



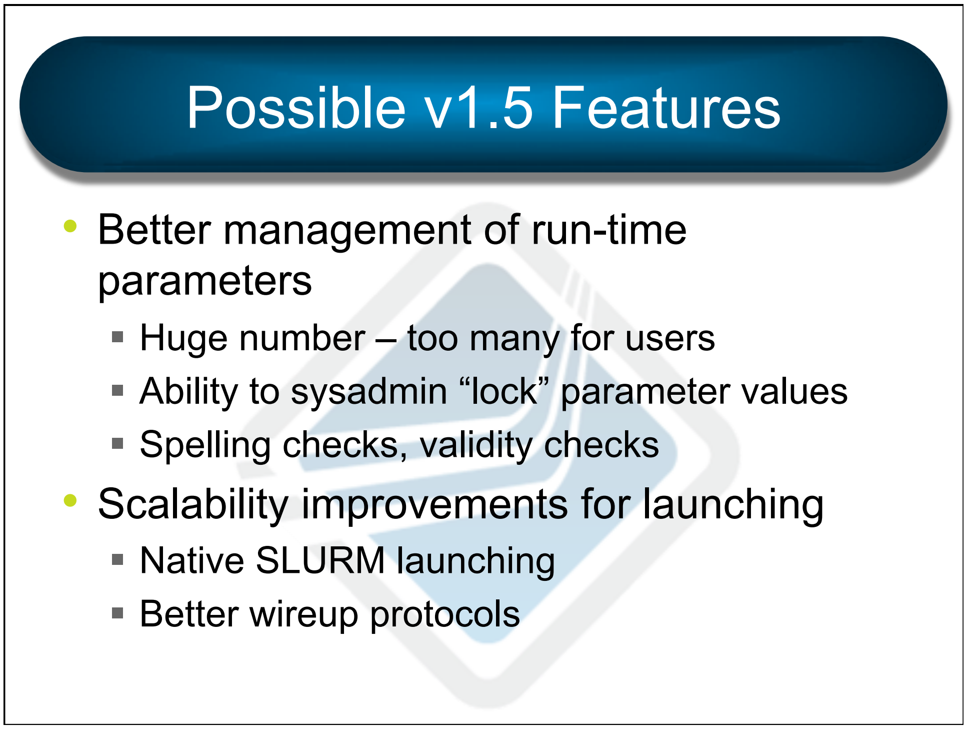
Logistics

- v1.5 → v1.6 series
- Release managers
  - Rainer Keller, Oak Ridge National Labs
  - Jeff Squyres, Cisco Systems
- Gatekeeper
  - George Bosilca, U. Tennessee



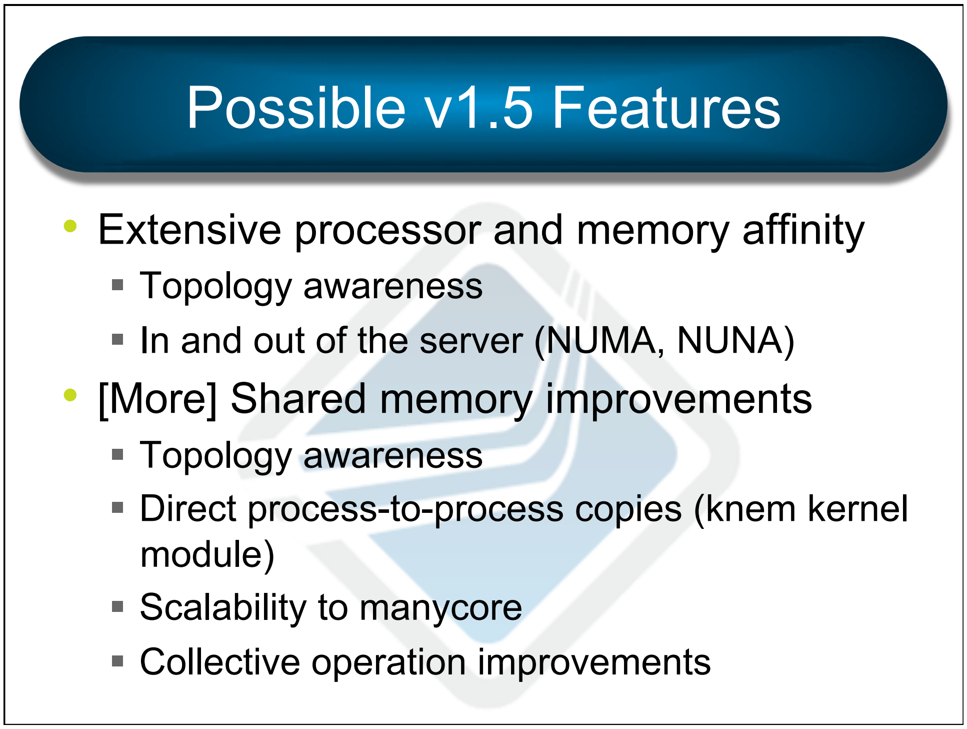
Possible v1.5 Features

- **BIG** disclaimer
  - Features discussed here are *possible*
  - “Nothing is decided until it is released”
- Not seeing something you want?
  - We’d love to see your patches ☺
- Full and updated list is on the OMPI Trac / wiki
  - Now accepting external accounts



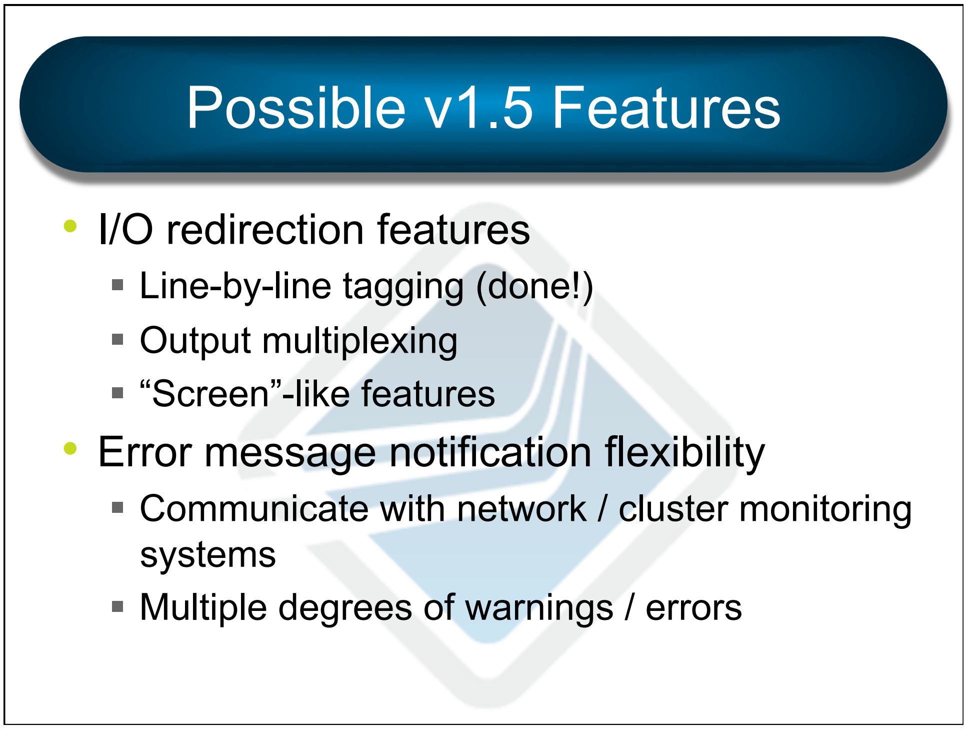
Possible v1.5 Features

- Better management of run-time parameters
  - Huge number – too many for users
  - Ability to sysadmin “lock” parameter values
  - Spelling checks, validity checks
- Scalability improvements for launching
  - Native SLURM launching
  - Better wireup protocols



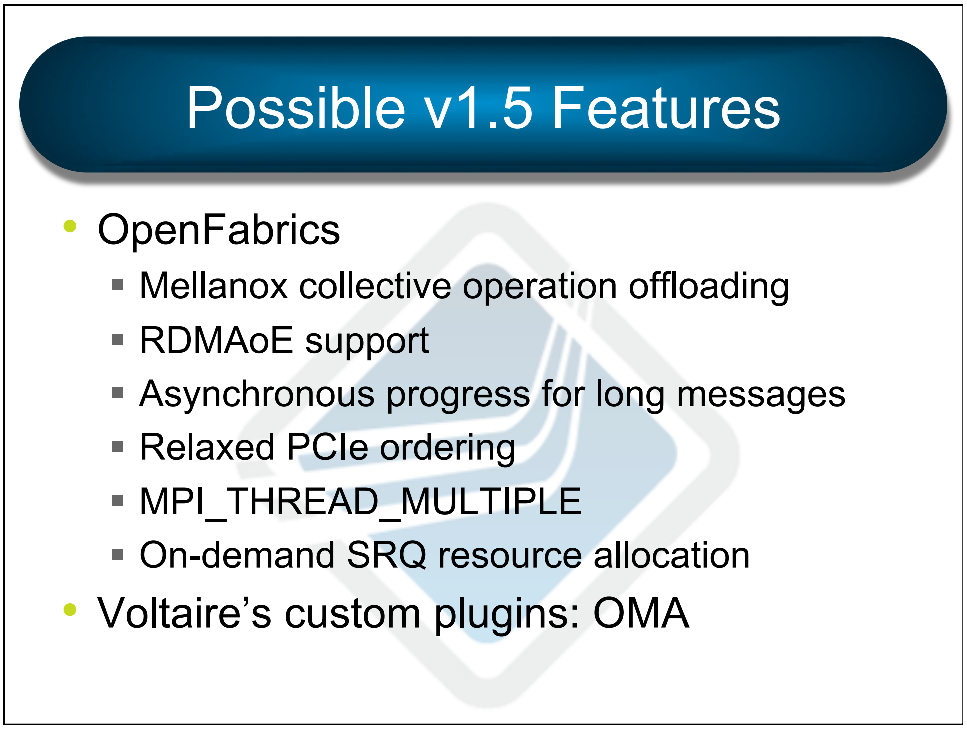
Possible v1.5 Features

- Extensive processor and memory affinity
  - Topology awareness
  - In and out of the server (NUMA, NUNA)
- [More] Shared memory improvements
  - Topology awareness
  - Direct process-to-process copies (knem kernel module)
  - Scalability to manycore
  - Collective operation improvements



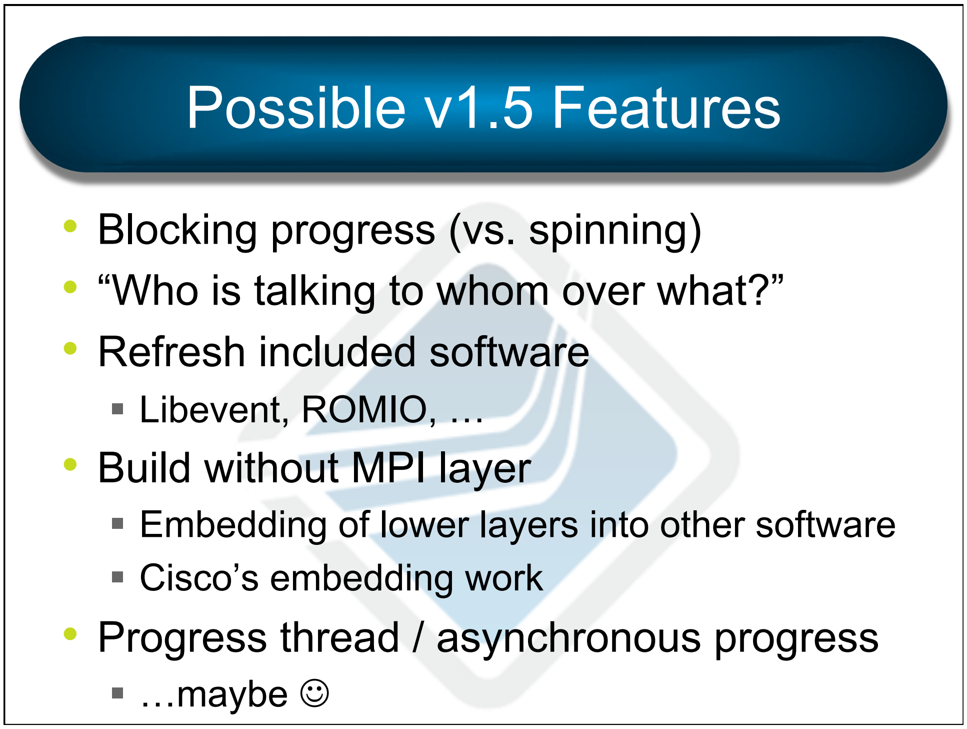
Possible v1.5 Features

- I/O redirection features
  - Line-by-line tagging (done!)
  - Output multiplexing
  - “Screen”-like features
- Error message notification flexibility
  - Communicate with network / cluster monitoring systems
  - Multiple degrees of warnings / errors



Possible v1.5 Features

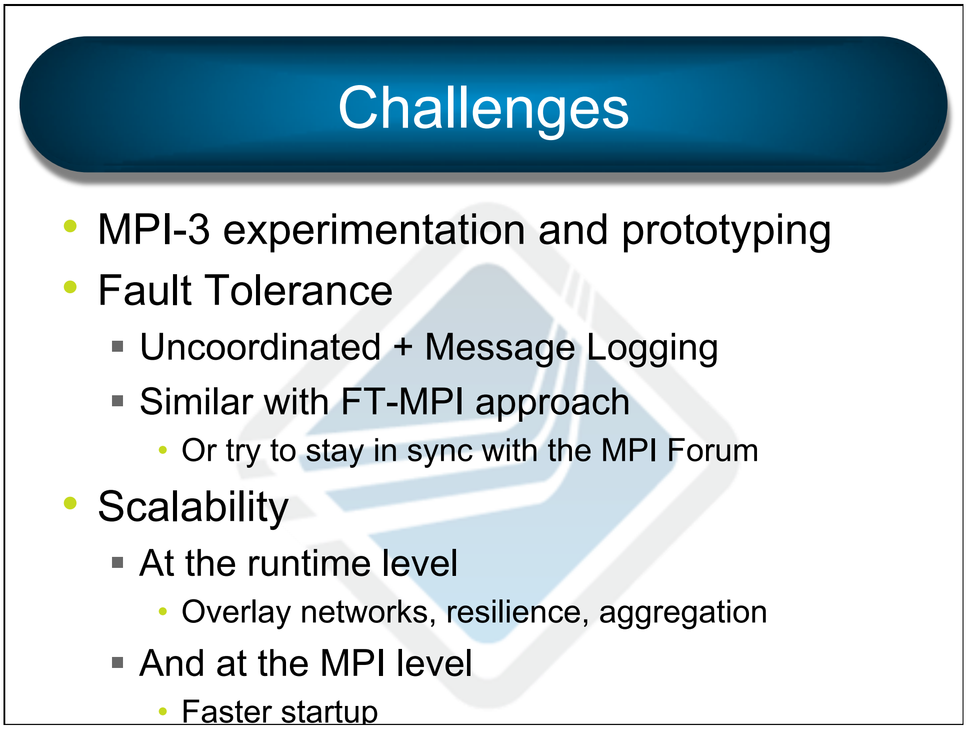
- OpenFabrics
  - Mellanox collective operation offloading
  - RDMAoE support
  - Asynchronous progress for long messages
  - Relaxed PCIe ordering
  - MPI_THREAD_MULTIPLE
  - On-demand SRQ resource allocation
- Voltaire’s custom plugins: OMA



Possible v1.5 Features

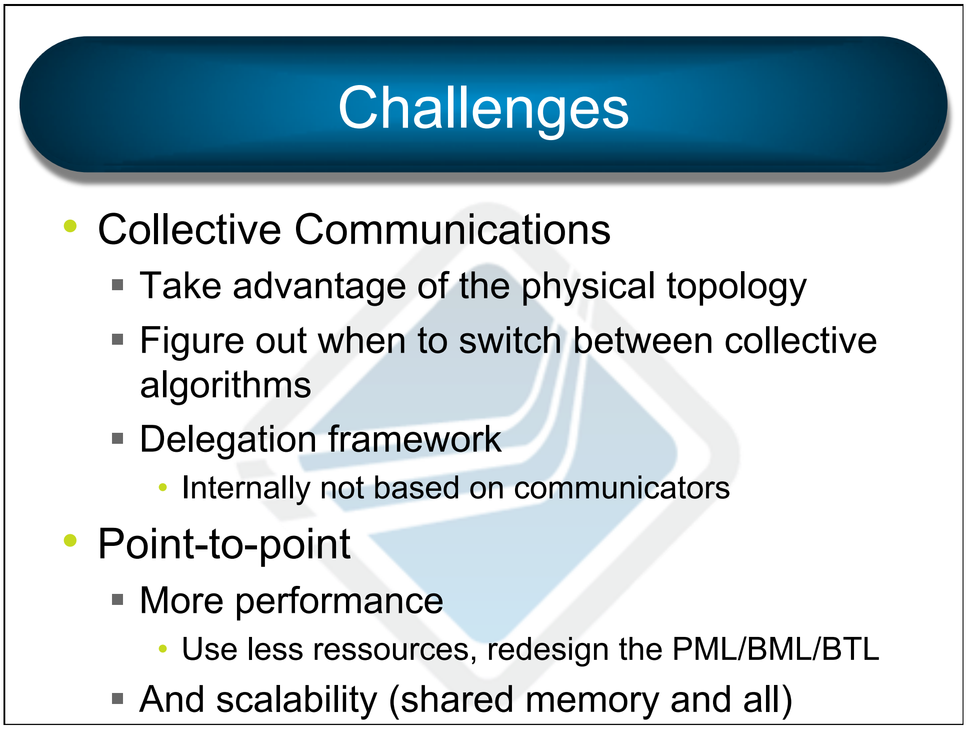
- Blocking progress (vs. spinning)
- “Who is talking to whom over what?”
- Refresh included software
  - Libevent, ROMIO, …
- Build without MPI layer
  - Embedding of lower layers into other software
  - Cisco’s embedding work
- Progress thread / asynchronous progress
  - …maybe ☺

Upcoming Challenges



Challenges

- MPI-3 experimentation and prototyping
- Fault Tolerance
  - Uncoordinated + Message Logging
  - Similar with FT-MPI approach
    - Or try to stay in sync with the MPI Forum
- Scalability
  - At the runtime level
    - Overlay networks, resilience, aggregation
  - And at the MPI level
    - Faster startup



Challenges

- Collective Communications
  - Take advantage of the physical topology
  - Figure out when to switch between collective algorithms
  - Delegation framework
    - Internally not based on communicators
- Point-to-point
  - More performance
    - Use less ressources, redesign the PML/BML/BTL
  - And scalability (shared memory and all)

HPC Community Feedback

Aside: MPI-2 Books

- MPI-2.2 is complete
  - $25 printed books (647 pages, $0.04/page!)
  - Take it home with you! ☺
  - **HLRS booth #2245**
- The MPI Forum wants your feedback
  - MPI-3 BOF session
  - Wednesday, 5:30pm, D-135

What do You Want From Open MPI?

Franklin D. Roosevelt:

Be sincere; be brief; be seated.

(we’re listening; you talk now)

How Important Is…

- Thread safety
  - MPI_THREAD_MULTIPLE
- Parallel I/O
  - Working with parallel file systems (which?)
  - ROMIO support ok?
- Dynamic processes
  - Spawn, connect / accept (anyone?)
- One-sided operations (MPI-3 revamp?)
  - Put, get, accumulate

Come Join Us!

http://www.open-mpi.org/

CISCO

ICL UT